
**Industrial automation systems
and integration — Product data
representation and exchange —**

Part 1:
Overview and fundamental principles

*Systèmes d'automatisation industrielle et intégration —
Représentation et échange de données de produits —*

Partie 1: Aperçu et principes fondamentaux

STANDARDSISO.COM : Click to view the full PDF of ISO 10303-1:2021



STANDARDSISO.COM : Click to view the full PDF of ISO 10303-1:2021



COPYRIGHT PROTECTED DOCUMENT

© ISO 2021

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

	Page
Foreword	v
Introduction	vi
1 Scope	1
2 Normative references	2
3 Terms, definitions and abbreviated terms	2
3.1 Terms and definitions	2
3.2 Abbreviated terms	8
4 Overview of ISO 10303	9
4.1 Purpose	9
4.2 Scope of ISO 10303	9
4.3 Fundamental principles	9
4.3.1 General	9
4.3.2 Integrated resources	9
4.3.3 Support for applications	10
4.3.4 Implementation methods	10
4.3.5 Implementations	10
4.3.6 Conformance testing	10
5 Architecture of ISO 10303	11
5.1 Concepts behind the architecture	11
5.2 Elaborations of the architecture	11
5.2.1 General	11
5.2.2 Sharing interpretations in the initial architecture	12
5.2.3 Sharing interpretations in the Modular architecture	12
6 Structure of ISO 10303	13
6.1 Series of parts	13
6.2 Description methods	13
6.2.1 Purpose	13
6.2.2 The EXPRESS and EXPRESS-G modeling languages (ISO 10303-11)	14
6.2.3 The EXPRESS-X language	14
6.3 Implementation methods	14
6.3.1 Purpose	14
6.3.2 Use of formal language	14
6.3.3 Mapping from EXPRESS to implementation method	14
6.3.4 Exchange structures for product data (implementation)	14
6.3.5 Standard data access interface specification	15
6.3.6 EXPRESS to OMG XMI binding	15
6.3.7 BO Model implementation	16
6.4 Application interpreted construct	16
6.4.1 Purpose	16
6.4.2 Characteristics	16
6.5 Application modules	16
6.5.1 Purpose	16
6.5.2 Characteristics	16
6.5.3 Business benefits	17
6.6 Business object models	17
6.6.1 Purpose	17
6.6.2 EXPRESS Model of the Business Object	17
6.6.3 An API Signature for a Business Object	17
6.6.4 A UML Model of the Business Object	18
6.7 Application protocols	18
6.7.1 Purpose	18
6.7.2 Definition of information requirements	18

6.7.3	Information representation	18
6.7.4	Implementation methods.....	18
6.7.5	Conformance requirements	18
6.8	Usage guides.....	18
6.8.1	Purpose.....	18
6.8.2	Characteristics	19
6.8.3	Document structure	19
6.8.4	Content.....	19
6.9	Conformance testing methodology and framework.....	19
6.9.1	Purpose.....	19
6.9.2	Procedures for conformance testing	19
6.9.3	Abstract test methods	20
6.10	Abstract test suites	20
7	Information object registration scheme	20
Annex A (normative)	Information object registration	22
Bibliography		23

STANDARDSISO.COM : Click to view the full PDF of ISO 10303-1:2021

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 184, *Automation systems and integration*, Subcommittee SC 4, *Industrial data*.

This second edition cancels and replaces the first edition (ISO 10303-1:1994), which has been technically revised.

The main changes compared to the previous edition are as follows:

- inclusion of application interpreted constructs;
- inclusion of application modules;
- inclusion of business object models;
- inclusion of additional implementation methods;
- inclusion of modular architecture as a preferred alternative to the initial architecture of ISO 10303-1:1994;
- permission for integrated resources to reference constructs written using EXPRESS from other International Standards;
- corrections to part numbering scheme;
- extension of information object registration to be usable on other standards;
- inclusion of usage guides;
- additional definitions.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

A list of all parts in the ISO 10303 series can be found on the ISO website.

Introduction

ISO 10303 is an International Standard for the computer-interpretable representation of product information and for the exchange of product data. The objective is to provide a neutral mechanism capable of describing products throughout their life cycle. This mechanism is suitable not only for neutral file exchange, but also as a basis for implementing and sharing product databases, and as a basis for archiving. The information generated about a product during its design, manufacture, use, maintenance, and disposal is used for many purposes. The use can involve many computer systems, including some that can be located in different organizations. In order to support such uses, organizations need to be able to represent their product information in a common computer-interpretable form that is required to remain complete and consistent when exchanged among different computer systems.

This document is an overview of ISO 10303. It specifies the overall scope of ISO 10303 and describes the ISO 10303 architectures and structure. It describes the various series of parts of ISO 10303 and the relationships among them.

ISO 10303 is organized as a series of parts, each published separately. The structure of ISO 10303 is described in this document.

Each part of ISO 10303 is a member of one of the following series: description methods, implementation methods, conformance testing methodology and framework, integrated generic resources, integrated application resources, application protocols, abstract test suites, application interpreted constructs, application modules and business object models.

A list of all parts in the ISO 10303 series is available from the following URL:

http://standards.iso.org/iso/10303/tech/step_titles.htm.

Industrial automation systems and integration — Product data representation and exchange —

Part 1: Overview and fundamental principles

1 Scope

This document provides an overview of ISO 10303.

ISO 10303 provides a representation of product information along with the necessary mechanisms and definitions to enable product data to be exchanged. The exchange is among different computer systems and environments associated with the complete product lifecycle, including product design, manufacture, use, maintenance, and final disposition of the product.

This document defines the basic principles of product information representation and exchange used in ISO 10303. It specifies the characteristics of the various series of parts of ISO 10303 and the relationships among them.

The following are within the scope of this document:

- scope statement for ISO 10303 as a whole;
- overview of ISO 10303;
- architectures of ISO 10303;
- structure of ISO 10303;
- terms and definitions used throughout ISO 10303;
- overview of data specification methods used in ISO 10303;

NOTE This includes the EXPRESS data specification language and graphical presentation of product information models.

- introduction to the series of parts of ISO 10303:
 - integrated resources;
 - application interpreted constructs;
 - application modules;
 - business object models;
 - application protocols;
 - implementation methods;
 - usage guides;
 - conformance testing methodology and framework;

- abstract test suites;
- scheme for identification of schemas and other information objects defined within parts of ISO 10303.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 8824-1, *Information technology — Abstract Syntax Notation One (ASN.1) — Part 1: Specification of basic notation*

3 Terms, definitions and abbreviated terms

3.1 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

3.1.1

abstract test case

ATC

specification, encapsulating at least one test purpose, that provides the formal basis from which executable test cases are derived and that is independent of both the implementation and the values

3.1.2

abstract test method

ATM

description of how an implementation is tested, given at the appropriate level of abstraction to make the description independent of any specific implementation of testing tools or procedures, but with sufficient detail to enable these tools and procedures to be produced

3.1.3

abstract test suite

ATS

part of ISO 10303 that contains the set of abstract test cases necessary for conformance testing of an implementation of an application protocol

3.1.4

agreement of common understanding

result of discussions between the partners of product data exchange or sharing, that ensures that all of them have the same understanding of the transferred or shared information

3.1.5

application

one or more processes creating or using product data

3.1.6

application activity model

AAM

model that describes an application in terms of its processes and information flows

3.1.7

application construct

collection of EXPRESS language entities, types, functions, rules and references that are based on resource constructs and that specialize those resource constructs as necessary to define a valid description of an aspect of product data for specific application areas

3.1.8**application context**

subset of an application activity model

3.1.9**application interpreted construct**

AIC

logical grouping of interpreted constructs that supports a specific function for the usage of product data across multiple application contexts

Note 1 to entry: See also *interpretation* ([3.1.41](#)).

3.1.10**application interpreted model**

AIM

information model that includes the application constructs necessary to satisfy the requirements of an application reference model

3.1.11**application module**

AM

reusable collection of a scope statement, information requirements, mappings and module interpreted model that supports a specific usage of product data across multiple application contexts

3.1.12**application object**

AO

atomic element of an application reference model that defines a unique concept of the application and contains attributes specifying the data elements of the object

3.1.13**application programming interface**

API

set of standard software interrupts, calls, functions, and data formats that can be used by an application program to access network services, devices, or operating systems

3.1.14**application programming interface implementation**

API implementation

implementation of ISO 10303 that supplies the services of the application programming interface, in contrast to an implementation of this standard that uses the API implementation

3.1.15**application programming interface instance**

API instance

individual execution context and state of an application programming interface implementation

Note 1 to entry: The notion of “execution context” in ISO 10303 is the same as in ECMAScript.

3.1.16**application programming interface signature**

APIS

description of the set of functions, protocols, and tools that programmers can use to enhance applications to access network services, devices, or operating systems

3.1.17

application protocol

AP
part of ISO 10303 that specifies an application interpreted model satisfying the scope and information requirements for a specific application

Note 1 to entry: This definition differs from the definition used in ISO 7498-2:1989 because the protocols address different contexts of use.

3.1.18

application reference model

ARM
information model that describes the information requirements and constraints of an application within an application protocol or module

3.1.19

application resource

integrated resource whose contents are related to a group of application contexts

3.1.20

assembly

product that is decomposable into a set of components or other assemblies from the perspective of a specific application

3.1.21

business object model

BO Model
single integrated information model for the scope of an AP

3.1.22

common resources

collection of information models, specified in the EXPRESS language, that can be reused to specify application-specific information models within the domain of industrial data

Note 1 to entry: The resource constructs defined by an application module are those defined in its module interpreted model schema.

Note 2 to entry: The term does not specify a specific series of ISO 10303 parts.

3.1.23

component

product that is not subject to decomposition from the perspective of a specific application

3.1.24

conformance assessment

analysis of conformance testing results to determine conformance to a particular application protocol's conformance class or option

3.1.25

conformance class

subset of an application protocol for which conformance can be claimed

3.1.26

conformance option requirement

incremental subset of an application protocol for which conformance can be claimed in addition to the conformance claimed for one or more conformance classes

3.1.27

conformance

precise, text definition of a characteristic required to be present in a conforming implementation

3.1.28**conformance testing**

testing of a conforming implementation by evaluating its data for the specific characteristics required by ISO 10303

3.1.29**data**

representation of information in a formal manner suitable for communication, interpretation, or processing by human beings or computers

3.1.30**data mapping language**

computer interpretable language that specifies how one information set relates to another

3.1.31**data exchange**

storing, accessing, transferring, and archiving of data

3.1.32**data specification language**

set of rules for defining data and their relationships suitable for communication, interpretation, or processing by computers

3.1.33**description method**

systematic procedure for specifying elements of a domain

EXAMPLE ISO 10303-11 is a description method that specifies the elements of the EXPRESS language.

3.1.34**digital signature**

result of a cryptographic transformation of data that, when properly implemented, provides a mechanism for verifying origin authentication, data integrity and signatory non-repudiation

3.1.35**domain**

area of knowledge, influence, or activity

3.1.36**exchange structure**

computer-interpretable format used for storing, accessing, transferring, and archiving data

3.1.37**externally defined**

explicit identification of the definition given within another source

Note 1 to entry: That source provides all the information necessary to unambiguously recreate the element; the information can be referenced rather than recreated.

3.1.38**generic resource**

integrated resource whose contents are independent of a specific application

EXAMPLE ISO 10303-42.

3.1.39**implementation method**

part of ISO 10303 that specifies a technique used by computer systems to exchange product data that is described using the EXPRESS data specification language

**3.1.40
implementation under test**

IUT

part of an application that is studied under testing

Note 1 to entry: This is normally an implementation of one or more characteristics of the standard(s) based on a given implementation method.

**3.1.41
information**

facts, concepts, or instructions

**3.1.42
information model**

formal model of information

Note 1 to entry: In ISO 10303, an information model is based on the Object-relationship modeling technique that organizes the product data as represented in different system aspects.

Note 2 to entry: In ISO 10303, information models are developed using EXPRESS modeling language.

EXAMPLE ISO 10303-242.

**3.1.43
integrated resource**

IR

part of ISO 10303 that defines a group of resource constructs used as the basis for product data. It includes the two types of resource parts: generic resources and application resources

Note 1 to entry: The 4x to 6x numbering is reserved for integrated generic resources and the 1xx numbering is reserved for integrated application resources.

EXAMPLE 1 ISO 10303-42.

EXAMPLE 2 ISO 10303-104.

**3.1.44
interpretation**

process of adapting a resource construct to satisfy an application-specific requirement of an application protocol

Note 1 to entry: The interpretation process can involve the addition of restrictions on attributes, the addition of constraints, and the addition of assignments.

**3.1.45
module interpreted model**

MIM

information model that uses the common resources necessary to satisfy the information requirements and constraints of an application reference model, within an application module

Note 1 to entry: The term common resources is not meant to imply that all such information models are required to be used in a module interpreted model regardless of domain or application.

EXAMPLE Three-dimensional geometry information models are common resources used in many MIMs. However, an application module describing colour will not use three-dimensional geometry information models as a resource.

**3.1.46
neutral**

independent of a specific computer system

3.1.47**predefined**

explicit description of product data given within a clause of a part of ISO 10303

Note 1 to entry: The explicit description provides all the information necessary to unambiguously recreate the element.

3.1.48**presentation**

recognizable visual representation of product data intended for use by humans

3.1.49**product**

thing or substance or information produced by a process

Note 1 to entry: Note to entry: In ISO 10303, it is common to include a product that is not a physical thing or substance, but that is information (e.g., a computer-interpretable model).

3.1.50**product data**

data that is a representation of product information

3.1.51**product information**

information about a product

Note 1 to entry: Note to entry: In ISO 10303, it is common to describe product information with respect to a specific industrial domain.

3.1.52**product information model**

formal model of product information

3.1.53**protocol implementation conformance statement**

PICS

statement of which capabilities and options are supported within an implementation of a given standard

Note 1 to entry: This statement is produced by completing a PICS proforma.

3.1.54**protocol implementation conformance statement proforma**

PICS proforma

standardized document in the form of a questionnaire, which, when completed for a specific implementation, becomes the protocol implementation conformance statement

3.1.55**resource construct**

collection of EXPRESS language entities, types, functions, rules and references that together define a valid description of an aspect of product data

3.1.56**standard data access interface**

SDAI

abstract Application Programming Interface (API) to work on application data according to a given data model defined in EXPRESS SDAI is specified in ISO 10303-22

Note 1 to entry: SDAI, which is specified in ISO 10303-22, is intended to be independent of programming language used.

3.1.57

STEP modules and resource library

SMRL

collection of application modules, business object models, and integrated resources intended to support one or more application protocols

3.1.58

Structural data mapping language

language elements that allow an unambiguous specification of a relationship between two schemas

Note 1 to entry: The two schemas both conform to the same schema definition language.

Note 2 to entry: The language elements of the structural data mapping language reference language elements of the schema definition language.

3.1.59

structure

set of interrelated parts of any complex thing, and the relationships between them

3.1.60

test purpose

precise description of an objective which an abstract test case is designed to achieve

3.1.61

unit of functionality

UOF

collection of application objects and their relationships that defines one or more concepts within the application context such that removal of any component would render the concepts incomplete or ambiguous

3.1.62

usage of product data

subset of the information flows in an application activity model

3.1.63

usage guide

specification that is a recommendation on applying an application protocol for a specific application domain within the scope of that application protocol

3.1.64

verdict

statement of "pass", "fail", or "inconclusive" concerning conformance of an implementation under test with respect to an executable test case and the abstract test case from which it was derived

3.1.65

verdict criteria

information defined within an abstract test case which enables the testing laboratory to assign a verdict

3.2 Abbreviated terms

DTD	document type definition
HDF	Hierarchical data format
STEP	Standard for the exchange of product model data
URL	Uniform resource locator
UML	Unified modeling language

XMI	XML metadata interchange
XML	Extensible mark-up language
XLSX	Microsoft Excel open XML spreadsheet

4 Overview of ISO 10303

4.1 Purpose

The purpose of ISO 10303 is to specify a form for the unambiguous representation and exchange of computer-interpretable product data throughout the life of a product. This form is independent of any specific computer system. This form enables consistent implementations across multiple applications and systems. ISO 10303 permits different implementation methods to be used for storing, accessing, transferring, and archiving product data. ISO 10303 defines a rigorous process for testing implementations for conformance.

4.2 Scope of ISO 10303

ISO 10303 provides a representation of product information along with the necessary mechanisms and definitions to enable product data to be exchanged. The exchange is among different computer systems and environments associated with the complete product lifecycle, including product design, manufacture, use, maintenance, and final disposition.

The following are within the scope of ISO 10303:

- representation of product information, including components and assemblies;
- implementations methods for exchange of product data, including storing, transferring, accessing, and archiving.

4.3 Fundamental principles

4.3.1 General

ISO 10303 separates the techniques of representation of product information from the implementation methods used for data exchange.

The representation techniques provide a single representation of product information common to many applications. This common representation may be adapted to meet the needs of specific applications. An application protocol (AP) specifies the representation of product information for one or more applications.

ISO 10303 specifies the implementation methods that support the exchange of product data defined in APs.

ISO 10303 defines a formal data specification language, EXPRESS which is used to specify the representation of product information. The use of a formal language provides unambiguous and consistent representation and facilitates development of implementations. The representation of product information may be annotated with a literal definition for each construct.

ISO 10303 provides a methodology and framework for conformance testing of implementations.

4.3.2 Integrated resources

A set of integrated resources (IRs) shall provide the specification of a representation of product information. Each IR comprises a set of descriptions, written in a formal data specification language,

applicable to product data known as resource constructs. One set may be dependent on other sets for its definition. A single resource construct may represent similar information for different applications.

The IRs in ISO 10303 are divided into two groups: generic resources and application resources. The generic resources are independent of applications and may reference other resources. The application resources may reference other resources and may add other resource constructs for use by a group of similar applications. The IRs may reference product data descriptions written using EXPRESS from other International Standards.

4.3.3 Support for applications

The IRs define a generic information model for product information. They are not sufficient to support the information requirements of an application without the addition of application specific constraints, relationships, and attributes.

ISO 10303 defines APs in which the IRs are interpreted to meet the product information requirements of specific applications. The interpretation is achieved by selecting appropriate resource constructs and refining their meaning, by specifying any appropriate constraints, relationships, and attributes. This interpretation results in an application interpreted model (AIM). The AIM is documented as part of an AP.

ISO 10303 has two mechanisms to ensure consistent interpretation when a resource construct represents the same information requirement in different APs. One mechanism uses interpreted resource constructs appearing in more than one AP as a documented application interpreted construct (AIC). The other mechanism uses application modules (AMs) that document the harmonized requirements along with the interpreted resource constructs.

NOTE [Subclauses 6.4](#) and [6.5](#) discuss application interpreted constructs and application modules in more detail.

The scope and information requirements of the application are specified using the terminology of the application domain. The AP provides a mapping to show how the interpretation of the IRs is used to meet the information requirements of the application domain.

ISO 10303 provides business object models that present complex models in a form that may be more understandable to application experts.

4.3.4 Implementation methods

Each implementation method included in ISO 10303 is specified by a mapping from the EXPRESS language onto the formal language used for the method. The mapping is independent of the AP. The mapping is expressed in a formal notation. At least three implementation methods for ISO 10303 have been identified (see [6.3.4](#)).

4.3.5 Implementations

An AP may specify one or more applicable implementation methods from the set of implementation methods in ISO 10303. An implementation shall apply one or more of the implementation methods specified in the AP to the AIM.

4.3.6 Conformance testing

Conformance of an implementation to an AP is specified by the conformance requirements in the AP.

A set of tests, specified in an abstract test suite (ATS), may be defined for each AP. When associated with an abstract test method (ATM), these tests can be used to assess the conformance of an implementation. The overall framework for conformance assessment is specified in ISO 10303-31.

An ATM for each implementation method is specified in one of the conformance testing methodology and framework series of parts of ISO 10303.

The scope of conformance testing of a specific implementation is the requirements specified for the conformance classes claimed for the implementation in the protocol implementation conformance statement (PICS). The tests are selected from the ATS, based on the conformance class claimed by the implementation. Test results provide the basis for conformance assessment.

ISO 10303 contains ATSs and defines ATMs for performing conformance testing to provide a basis for test results that are repeatable, comparable, and auditable. Inclusion of conformance testing procedures in ISO 10303 is intended to promote widespread acceptance of test results.

5 Architecture of ISO 10303

5.1 Concepts behind the architecture

The ISO 10303 architecture is designed to support the development of standards for product data exchange and product data sharing. The architecture is constrained by the following concepts:

- the scope of what is standardized and what is conformance tested is set at the level of an application;
- Information requirements are based on a model of a business activity;
- information requirements are standardized using natural language; and
- a mapping is specified that defines the data structures and population constraints using the IRs that satisfy the information requirements.

ISO 10303 is implemented through APs. An AP consists of the following major elements:

- an AAM describing the business process that the information model supports;
- an ARM specifying the information requirements;
- an information model based on the IRs, called an application interpreted model (AIM), that is a basis for implementations of ISO 10303;
- zero or more information models for data structures based on the ARM, called a business object model (BO Model), that is a basis for implementations of ISO 10303.

The architecture is based on standardizing industry information requirements and mapping those information requirements to an information model based on the IRs. The process of mapping information requirements to the IRs is known as "interpretation".

The ISO 10303 architecture focuses on the information required by industrial processes rather than on the processes themselves, as the processes can change over time, while the underlying information requirements are longer lasting. This focus on information allows ISO 10303 to support data exchange, some forms of data sharing, as well as long-term data retention.

Whenever a resource construct is used to represent the same information requirement in different APs, the same interpretation of that resource construct shall be used.

5.2 Elaborations of the architecture

5.2.1 General

The general architectural principles described in 5.1 have been elaborated into two architectures: the "initial architecture" and the "modular architecture". Both architectures adhere to the fundamental principles described in 4.3. The primary change introduced by the modular architecture is the explicit harmonization of common information requirements.

NOTE The term "ISO 10303 architecture", without qualification, refers to the general principles described in 5.1 and the common aspects of its elaborations.

An ISO 10303 project may use either the initial architecture or the modular architecture, although new projects are encouraged to use the modular architecture. ISO TC 184/SC 4 shall determine, at the time a project is approved, which architecture the project will use.

5.2.2 Sharing interpretations in the initial architecture

An AIC shall be used to specify a piece of an AIM that can be used to exchange product data common to two or more application protocols.

NOTE An AIC does not document the common information requirements or the mapping of those information requirements into the AIM.

An AP that complies with initial architecture is called a "monolithic AP".

5.2.3 Sharing interpretations in the Modular architecture

5.2.3.1 General

Rather than relying on harmonization occurring as a by-product of consistent interpretation across APs, application requirements are first harmonized across domains and the resulting mappings are standardized in AMs. AMs are reused by other AMs and ultimately by APs.

The implementable portion of the modular architecture has two core components and one optional component:

- AM: A small, reusable data specification documented with an ARM; mapping; MIM; and usage guide.
- AP: The use of a data specification to meet the requirements of some business process.
- BO Model: The use of a data specification that is optimized for implementation.

NOTE The BO Model is an optional component.

The objectives and function of the architectural components are described below.

5.2.3.2 Application module

The design of an application module should maximize reusability of the:

- a) harmonized requirements,
- b) associated interpretation into the IRs,
- c) data specification, and
- d) software implementations.

The design supports reusability by the standards developer, implementer, and user.

AMs replace AICs in the modular architecture. The objectives of AICs and AMs are similar. They both standardize interpretation results for reuse in multiple APs. However, AICs and AMs are created differently and have different content. AMs, unlike AICs, contain harmonized information requirements and specifications of the mappings of those requirements to the IRs. An objective of modularization is to document a concept one time and then to directly reuse that concept in other AMs. An AIC is only created when a concept has already been documented in two or more APs.

The modular architecture uses EXPRESS rather than natural language for the documentation of requirements, that is, the ARM, in an AM. This allows the use of tools to validate the dependencies between AM ARMs.

5.2.3.3 Modular application protocol

An AP that complies with modular architecture is called a "modular AP". A modular AP is a documented use of a set of application modules (AMs) for a specific business process. The AMs used by an AP are organized in a tree structure. A single, normatively referenced AP application module (AP AM) is the data specification for the AP. This AM normatively references a collection of related AMs that provide the documentation of its information requirements and standardized interpretations. The AP AM can include specific business process rules or constraints. The AP document contains an activity model and conformance class definitions. Industry specific terminology may be mapped to the generic AM terminology defined in an AP to make it more understandable to reviewers from the application domain.

5.2.3.4 Business object model

A business object model (see 6.6) may be specified in languages such as EXPRESS, UML, or XML Schema. Business objects may aggregate lower-level objects from the primary requirements model to hide data complexity or may be augmented with additional data to complete the high-level concepts upon which the business objects are based. A business object model implementation form shall be in XML, with the appropriate configuration specification for use with 10303-28 specified in an Annex for the derivation of the XML format specification from the EXPRESS representation of the business object model.

6 Structure of ISO 10303

6.1 Series of parts

ISO 10303 is divided into the following series of parts. Each series has a unique function and consists of one or more parts. The series are listed below with their numbering scheme:

- description methods: parts 11 to 19;
- implementation methods: parts 21 to 29;
- conformance testing methodology and framework: parts 31 to 39;
- integrated resources:
 - generic resources: parts 41 to 99;
 - application resources: parts 101 to 199;
- application protocols: parts 201 to 299;
- abstract test suites: parts 301 to 399 (corresponding to the associated APs 201 to 299);
- application protocol modules: parts 401 to 499 (corresponding to the associated APs 201 to 299);
- application interpreted constructs: parts 501 to 599;
- application modules: parts 1001-1999;
- business object models: parts 3001 to 3099;
- usage guides: parts 5001 to 5999.

6.2 Description methods

6.2.1 Purpose

The description of product data in common resources and APs requires the use of formal data specification languages to ensure consistency and avoid ambiguity. The languages are intended to be

both human-readable to facilitate human understanding and computer-interpretable to facilitate the generation of application software and supporting tools.

6.2.2 The EXPRESS and EXPRESS-G modeling languages (ISO 10303-11)

EXPRESS is a formal modeling language that provides the mechanism for the normative description of product data for both common resources and APs.

EXPRESS allows a description of the data and constraints applicable to product data in a closed domain of discourse suitable for conformance assessment. EXPRESS permits the definition of resource constructs from data elements, constraints, relationships, rules and functions. The language permits classification and structuring of resource constructs. Resource constructs may be interpreted within APs. The interpretation capability of EXPRESS is a mechanism to facilitate the development of APs by allowing the addition of restrictions on attributes, the addition of constraints, the addition of relationships among resource constructs and application constructs, or all the above.

EXPRESS-G is a graphical representation for the structural constructs in the EXPRESS language to facilitate human understanding.

6.2.3 The EXPRESS-X language

EXPRESS-X is specified in ISO 10303-14 and is a structural data mapping language. It allows the unambiguous specification of relationship between data that is governed by EXPRESS schemas, and for specifying alternative views of data that is governed by EXPRESS schemas.

6.3 Implementation methods

6.3.1 Purpose

ISO 10303 provides for various methods of implementation. An implementation method provides a specific method for using the APs defined in ISO 10303. The exchange structure provides for the writing and reading of the description of product data in APs, using clear text or binary encoding.

EXAMPLE ISO 10303-21 specifies an implementation method for an exchange structure.

6.3.2 Use of formal language

An implementation method is defined using a formal language so that computer-based methods may be used for the development of implementations. Formal languages provide a precise description for conformance.

6.3.3 Mapping from EXPRESS to implementation method

EXPRESS models provide the basis for all specifications of product information in ISO 10303. Each implementation method defined in ISO 10303 specifies mapping rules from the syntax of EXPRESS onto the language used for the implementation method. The structure and syntax of each resource construct as it appears in the implementation is derived by applying the mapping rules. The rules to be applied for a specific implementation method may be conditional on the form of the EXPRESS definition that is being mapped. Any schema defined in EXPRESS can be mapped to the implementation method.

6.3.4 Exchange structures for product data (implementation)

ISO 10303 provides three different ways of exchanging product data described in EXPRESS between computer systems:

- clear text encoding of the exchange structure (ISO 10303-21);
- binary representation of EXPRESS-driven data (ISO/TS 10303-26);

— XML representations of EXPRESS schemas and data (ISO 10303-28).

ISO 10303-21 provides a compact and human readable form of the exchange structure. The exchange structure can be implemented to support file exchange. The language used for the syntax of the exchange structure implementation method is based on Wirth Syntax Notation. The mapping from EXPRESS to the syntax of the exchange structure is defined.

ISO/TS 10303-26 provides a mapping of EXPRESS to the binary Hierarchical Data Format Version 5 (HDF5) that is optimized for the exchange of huge data sets. HDF5 is a product of The HDF Group, a spinoff from the National Center for Supercomputing Applications (NCSA) at the University of Illinois at Urbana-Champaign. The HDF software includes I/O libraries and tools for analysing, visualizing, and converting scientific data.

ISO 10303-28 enables product data described in EXPRESS to be exchanged using XML and the many software tools developed to support XML technologies. It also permits product data sets so described to be readily incorporated into electronic commerce transactions represented in XML. ISO 10303-28 specifies the form of XML documents containing EXPRESS schemas and data governed by EXPRESS schemas. For an arbitrary EXPRESS schema, it specifies an XML schema that corresponds to the EXPRESS schema. It also contains a set of configuration directives that can be used to specify options for the structure of the XML representation of data sets that conform to EXPRESS schemas.

ISO 10303-28 specifies an XML document type definition (DTD) that corresponds to the EXPRESS schema. It can be used by implementers who need to use DTDs instead of XML schemas.

6.3.5 Standard data access interface specification

The standard data access interface (SDAI) specifies the functional characteristics of an application programming data access interface to product data. SDAI defines the operations available to an application for the purposes of accessing and manipulating data with a structure defined using EXPRESS. SDAI is specified in ISO 10303-22. The SDAI is defined independently of any programming language.

The specification of the SDAI functionality in a specific programming language is known as a language binding. A language binding provides access to and manipulation of data entities, types and constants, mechanisms for constraint validation and error handling. Language bindings are provided as separate parts of ISO 10303.

At the time of preparation of this document, the following language bindings are available:

- C++ language binding to the SDAI (ISO 10303-23);
- C language binding to the SDAI (ISO 10303-24);
- Java(tm) programming language binding to the SDAI with Internet/Intranet extensions (ISO/TS 10303-27).

Generic conformance testing for SDAI implementations is provided by ISO/TS 10303-35. For each language binding, these abstract test methods are converted into executable test methods for conformance testing.

6.3.6 EXPRESS to OMG XMI binding

ISO/TS 10303-25 specifies a mapping of EXPRESS constructs to the UML meta-model. Because the XMI standard specifies the XML representation of UML meta-model constructs, standardizing the mapping of EXPRESS constructs into UML constructs supports the XMI representation of EXPRESS schemas. EXPRESS schemas, being data specifications, are mapped into the UML meta-model concepts that appear in UML Static Structure Diagrams. ISO/TS 10303-25 does not map all EXPRESS constructs to the UML meta-model, because that meta-model does not support all the corresponding EXPRESS concepts. The specified mapping is a one-way mapping from EXPRESS into the UML Interchange Meta-model.

These limitations make the mapping unsuitable for the general interchange of EXPRESS schemas and UML models for information modelling purposes.

6.3.7 BO Model implementation

A BO Model may be implemented in any 10303 implementation form of an EXPRESS information model derived from another International Standard into the integrated resources. The derivation shall be approved by a resolution of ISO TC 184/SC 4. The IRs comprise a single information model with a single name space. Because such an EXPRESS model is included in an integrated resource model, it may be referenced by any other EXPRESS model in the integrated resources. Any EXPRESS models that it references shall be included as integrated resources, to ensure that the integrated resources constitute a self-contained model.

NOTE At the time of preparation of this document, the only such model is ISO 13584-20.

IRs provide a unique representation of each element of information within ISO 10303. The meanings of the resource constructs are provided through text definitions.

The IRs are divided into logically-related sets of one or more resource constructs. The generic resources are context-independent. The application resources are applicable to specified ranges of applications.

Generic and application resource constructs may reference generic resource constructs.

6.4 Application interpreted construct

6.4.1 Purpose

The purpose of an AIC is to provide a mechanism to identify and document the common requirements of different APs as represented within their AIMS. AICs provide a consistent and standardized interpretation of the IRs across different application contexts by identifying the semantics and shared data definitions to support a specific functionality.

6.4.2 Characteristics

AICs may be referenced by APs, AMs or other AICs in their entirety. In the modular architecture, typically one AIC is encapsulated in one AM.

6.5 Application modules

6.5.1 Purpose

AMs are the key constituent of the modular architecture. The basis of the modular approach is understanding and harmonizing the requirements, both new and those documented in existing APs, grouping the requirements into reusable modules, and using the modules in the development of an AP.

6.5.2 Characteristics

An AM contains much of the technical content that, under the initial architecture, was documented in an AP. The role of an AP document in the modular architecture is to provide a business context for the industrial use and implementation of the AMs that are the data specification of the AP.

There are three types of AMs: foundation modules (level 1), implementation modules (level 2), and AP modules (level 3).

Foundation modules provide lower-level reusable structures that are not likely to be implemented alone but are highly shareable and reusable. Implementation modules define a capability that can be implemented and against which conformance classes can be defined. Implementation modules may include AP specific data population constraints omitted from foundation modules. Each AP

references a single root module, which is called the "AP application module" (AP AM). An AP AM is an implementation module, and the contents of an AP AM are the same as other implementation modules, the only documentation difference being in their name and title. The AP module from one AP can be used by another AP. The AP module has a part number equal to the AP's part number plus 200.

EXAMPLE The AP module for AP ISO 10303-210 is ISO/TS 10303-410.

A detailed description of the content of an AM is provided in guidelines for the content of application protocols. A detailed description of the content of an AP that uses AMs is provided in Reference.

6.5.3 Business benefits

The modular architecture provides:

- the ability to implement a combination of subsets of multiple APs, or to extend existing APs to meet a business need;
- the ability to reuse application software developed to support one AP in the development of an implementation of another AP with the same, or similar, requirements;
- the ability to avoid the duplication and repeated documentation of the same requirements in different APs, leading to potentially different solutions for the same requirements;
- the ability to reuse data generated by an implementation of one or more APs by an implementation of one or more different APs;
- the ability to refactor a module into two or more AMs without changing the external application;
- the ability to significantly reduce maintenance cost for APs that share common subset.

6.6 Business object models

6.6.1 Purpose

A BO model specifies an information model at a high-level of granularity suited for communication of concepts to domain experts. The creation of a BO model begins with analysis of the ARM concepts. When the desired business object complexity warrants simplification, an aggregation of the ARM concepts is specified. Three different options for documentation to automate implementation are described in [6.6.2](#), [6.6.3](#) and [6.6.4](#).

EXAMPLE 1 A web-based interface for C++ programming is an automated implementation.

EXAMPLE 2 A database schema for a parts catalogue is an automated implementation.

NOTE 1 "Business Object Models for Industrial Data Standards" provides additional information about business object models.

NOTE 2 The three options are the methods used for implementations, not the implementations themselves.

6.6.2 EXPRESS Model of the Business Object.

This option provides some implementation automation when used with an implementation form such as XML Schema with a specified ISO 10303-28 configuration option.

6.6.3 An API Signature for a Business Object.

. This option addresses issues related to interface development based on the AIM. The information in the API signature is directly instantiated in the AIM. The signature is built on the AIM concepts that correspond to the ARM concepts that are related to the business object, along with any further processing requirements in a format suitable for an API in pseudo-code. It also enables the automatic generation of API functions in languages such as C++, early or late binding, and web services.

6.6.4 A UML Model of the Business Object.

This option provides implementation automation under the assumption that the associated XML Metadata Interchange would be used to automatically generate implementation forms such as web services.

6.7 Application protocols

6.7.1 Purpose

An AP specifies an information model, suitable not only for neutral file exchange, but also as a basis for implementing and sharing product databases and archiving, within an application domain. For a monolithic AP, the information model is specified as an integral part of the AP document. For a modular AP, the information model is specified by reference to the AP module.

6.7.2 Definition of information requirements

An AP includes the definitions of scope, context, and information requirements of an application. The definitions may specify functions, processes or information that are excluded from the application to clarify the scope, context, and information requirements. The statement of scope is supported by an AAM that describes the processes, information flows, and functional requirements of the application. The activity model is included as an informative annex to the AP.

The information requirements and constraints for the application context are defined by means of a set of application objects included in an ARM. This definition is derived from an ARM. An ARM is a formal information model that is documented in an informative annex to the AP.

6.7.3 Information representation

The AIM is assembled from resource constructs specified by the IRs. The resource constructs are interpreted to meet the application requirements within the defined context and scope of the AP.

A mapping from the information requirements to the AIM is provided. A mapping language is specified in guidelines for mapping specifications. The mapping defines the use, within the AIM, of resource constructs from the IRs to represent the information requirements of the application. For each ARM requirement, the mapping provides an unambiguous query of the AIM information base. A mapping from the information requirements to the AIM is provided in each AP.

6.7.4 Implementation methods

The AP is independent of implementation methods. However, an AP may include implementation method specific information in a normative annex. An AP includes reference to a table of short names that the exchange structure implementation method uses to encode AP entity names. The table is available at https://standards.iso.org/iso/10303/tech/short_names/short-names.txt

6.7.5 Conformance requirements

An AP includes the conformance requirements to be satisfied by any implementation claiming to support the requirements of the AP. Conformance requirements reflect the capabilities defined in an AP and may be specified in the description methods, implementation methods, or AP series of parts of ISO 10303.

6.8 Usage guides

6.8.1 Purpose

A usage guide contains guidance on implementing and using an AP. A usage guide may provide guidance to two different audiences: implementors and end users of AP-compliant implementations.

6.8.2 Characteristics

A usage guide may define refined context, scope, and information requirements for a subset of the context, scope, and information requirements of the subject AP, and specify the use of the AP's information representation to satisfy these requirements.

EXAMPLE 1 ISO/TS 10303-5001 defines the context, scope, and information requirements for various development phases during the design of a gear unit and specifies the ISO 10303-214 resources necessary to satisfy these requirements.

An AP may have more than one usage guide covering the same subject area, but for different disciplines.

EXAMPLE 2 ISO 10303-227 was developed jointly by the shipbuilding and process plant communities. There could be separate usage guides for ISO 10303-227 for ship piping and process plant piping, to document usage of the ISO 10303-227 information representation specific to the respective communities, or the software products they use.

6.8.3 Document structure

A usage guide may be included as an annex of an AP. One or more usage guides may be published as separate documents, in addition to or instead of such an annex.

6.8.4 Content

A usage guide may include its own information requirements and reference model, separate from the information requirements and ARM of the AP. If so, the usage guide shall include a mapping of its information requirements and reference model to those of the AP. The mapping may be to the AP's ARM or AIM, or there may be mappings to both ARM and AIM.

Example product descriptions that are supported by the AP and the corresponding AP exchange files can be included in a usage guide. If exchange files are included, the usage guide should explain the primary data structures and the logic and meaning of the values used in the exchange file.

6.9 Conformance testing methodology and framework

6.9.1 Purpose

The conformance testing methodology and framework series of ISO 10303 provides the general methodology and requirements for the process of testing the conformance of a product that claims to implement an ISO 10303 AP. The goal of the methodology and framework of conformance testing is to ensure:

- Repeatability: test results are consistent whenever undertaken;
- Comparability: test results are consistent wherever undertaken;
- Auditability: test procedures can be confirmed as having been correctly undertaken, following the testing, by review of record.

ISO 10303-31 provides a framework and describes the general concepts for conformance testing of implementations of ISO 10303.

6.9.2 Procedures for conformance testing

Conformance testing of an AP can be performed by applying the abstract test method for the chosen implementation method to the test cases in the abstract test suite. The procedures for conformance testing are independent of the implementation under test.

If a single implementation combines several APs, conformance testing is performed for each AP separately.